



## Code Inspection Checklist

**Number:** 580-CK-056-01  
**Effective Date:** April 25, 2006  
**Expiration Date:** April 25, 2011

**Approved By: (signature)**  
**Name:** Barbara Pfarr  
**Title:** Assoc. Chief, ISD

**Responsible Office:** 580/Information Systems Division (ISD)  
**Title:** Code Inspection Checklist

**Asset Type:** Checklist  
**PAL Number:** 2.4.2.1

### Code Inspection Checklist

The Code Inspection Checklist defines the criteria to be used during a code inspection for the review and approval of unit code. For a detailed explanation of how inspections are conducted, and how they differ from peer reviews and walkthroughs, please refer to "Inspections, Peer Reviews, and Walkthroughs," PAL #3.2.3. Note that a key purpose of the code inspection is to verify that the code complies with the detailed design. For additional discussion of detailed design, refer to the Detailed Design process, PAL #2.3.2, and the Design Inspection/Walkthrough Checklist, PAL #2.3.1.3.

This checklist should be used for all critical deliverable units, both new and revised; it need not be used for non-deliverable code. Note also that this checklist should be tailored for projects or organizations that use automated code verification tools.

For each checklist item below, place a check (✓) in the box if the checklist item is satisfied. Otherwise, list any problem areas or exceptions under "Issues and Comments."

		✓	Issues and Comments
1	<b>Compliance with standards</b> – Do the unit code, and unit test plan (or unit test cases) comply with ISD or tailored Branch/project-level standards and naming conventions?  <i>Guidance: If, for example, applicable standards require that the code include prologues and/or Program Design Language (PDL), then those should be included. If the standards require comments, then comments should be included.</i>  <i>Guidance: In cases where iterative development is being used, unit testing is sometimes performed prior to the code inspection. In such cases, the unit test results should also be examined during the code inspection.</i>  <i>Guidance: If any waivers to applicable standards have been granted, make a notation to this effect under "Issues and Comments." Approved ISD standards may be found on the EPG web site at <a href="http://software.gsfc.nasa.gov/process.cfm">http://software.gsfc.nasa.gov/process.cfm</a>.</i>	<input type="checkbox"/>	
2	<b>Overall correctness</b> – Does the unit do what it's supposed to do?	<input type="checkbox"/>	

3	<b>Consistency of Control Logic</b> – Is the control logic consistent between the design (e.g., PDL, design specifications, or class libraries) and the code? Has the unit design been converted correctly to code?  <i>Guidance: Note that in some tailored or streamlined versions of the software development life cycle, design and code phases may occur in parallel, or PDL may serve as pseudo-code. This checklist should be appropriately tailored in such cases.</i>	<input type="checkbox"/>	
4	<b>External Functions or Units Called</b> – Are all functions or units called in the code consistent with the functions or units named in the unit design (e.g., PDL, design specifications, or class libraries)?  <i>Guidance: In some instances, not all of the functions or units to be called may be known at unit design time.</i>	<input type="checkbox"/>	
5	<b>Correctness of Calling Parameters</b> – Are all functions and units called correctly? That is, do the correct calling parameters appear, in the correct order?	<input type="checkbox"/>	
6	<b>Clarity</b> – Is the purpose of all coding statements clear?	<input type="checkbox"/>	
7	<b>Commenting or Documentation</b> – Are internal comments or documentation statements included within the code?  <i>Guidance: Can a person unfamiliar with the code use the comments to understand and maintain the code?</i>	<input type="checkbox"/>	
8	<b>Use of debug statements</b> – Are any debug statements clearly set apart from, and distinguished from, nondebug statements?	<input type="checkbox"/>	
9	<b>Files and Databases Accessed</b> – Are all files and databases accessed in the code consistent with the files and databases named in the unit design?  <i>Guidance: Note that in some instances, not all of the precise files and databases to be used may be known at unit design time.</i>	<input type="checkbox"/>	
10	<b>Access of files and database records</b> – Are all external files or database records read or written correctly?  <i>Guidance: Verify that file or database read/write statements are consistent with the structure of the corresponding files or databases. Place a check in the box if all accesses to files or database records are coded correctly. List any exceptions under "Observations and Comments."</i>	<input type="checkbox"/>	
11	<b>Files Opened and Closed</b> – If the unit opens any external files, does it also close them? If not, is there a logical reason for not closing them?	<input type="checkbox"/>	

12	<b>Memory Allocated and Deallocated</b> – If the unit allocates any memory storage, does it also deallocate it? If not, is there a logical reason for not deallocating the memory?	<input type="checkbox"/>	
<b>Notes/Action Items for follow-up</b>			
#	Action	Assignee	Due Date

#### References

- SEAS System Development Methodology (SSDM), Computer Sciences Corporation, 1989.
- ALMRS/Modernization Project Quality Standards and Procedures, Computer Sciences Corporation, October 6, 1993.

#### Change History

Version	Date	Description of Improvements
1.0	4/24/06	Initial approved version by CCB